

FIG. 1

BEST AVAILABLE COPY

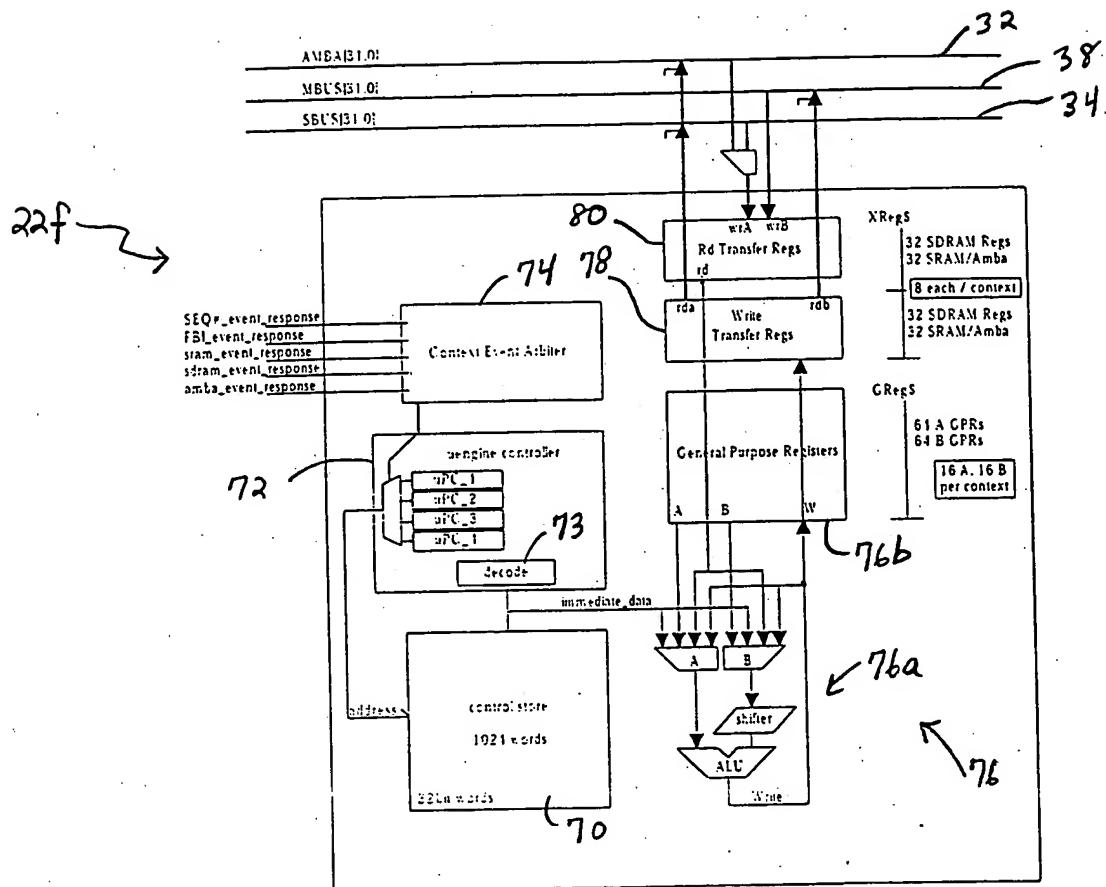
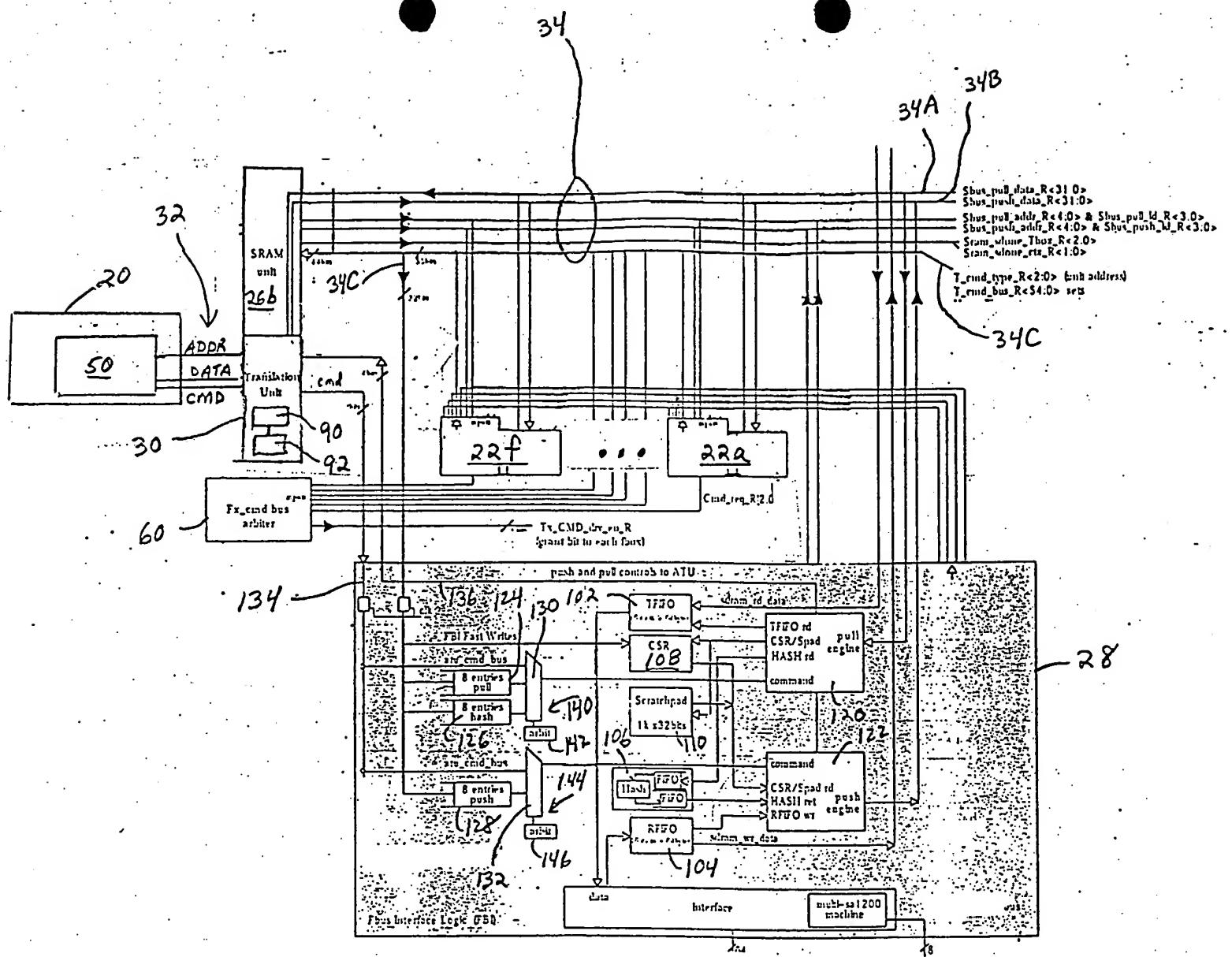


FIG. 2

BEST AVAILABLE COPY



Cmd\_Reg\_R<2:0>  
 000 none  
 001 Sram Chain  
 010 SDR Chain  
 011 Sram  
 100 SDR  
 101 FSI  
 110 PCI  
 111  
 Tx\_CMD\_crv\_en\_R<1:0>  
 0 none  
 1 grant

Sram\_puXX\_addr\_R<6:0> Sram\_puXX\_addr\_R<4:0>  
 [4:0] sref\_rts\_rts  
 4-bit TFFO  
 [6:0] TFFO\_addr  
 Sram\_puXX\_ID\_R<3:0> Sram\_puXX\_ID\_R<3:0>  
 3-5 Flaves  
 8-13 Flaves-csr  
 6-7 Flaves-csr  
 15 npt

T\_Cmd\_Type\_R<2:0>  
 000 : bus Rts  
 001 : SDRAM  
 010 : SDR  
 011 : SRAM-csr  
 100 : PCI  
 101 : reserved  
 110 : FFI  
 111 : Scratch

FIG. 3

BEST AVAILABLE COPY

ADDRESS SPACE  
(Hexadecimal)

FFFF FFFF	
E7FF FFFF	
E000 0000	
C000 0000	SDRAM
B000 0000	TRANSLATION UNIT 30
A000 0000	PCI CONFIG
9000 0000	SYSTEM
8000 0000	MAC & SRAM CONTROL REGISTERS
4000 0000	PCI UNIT
0000 0000	SRAM

Hexadecimal	Mapped Addresses	XXXX (Binary)
B004 XXXX	FBUS Interface (28) Scratchpad (110)	0100 xxxx xxxx xxxx
B004 XXXX	FBUS Interface (28) Registers (108)	0000 xxxx xxxx xxxx
B000 XXXX	Micro-engine (22a) Registers (78, 80)	0110 1xxx xxxx xxxx
	Micro-engine (22b) Registers (78, 80)	0110 0xxx xxxx xxxx
	Micro-engine (22c) Registers (78, 80)	0001 1xxx xxxx xxxx
	Micro-engine (22d) Registers (78, 70)	0001 0xxx xxxx xxxx
	Micro-engine (22e) Registers (78, 80)	0000 1xxx xxxx xxxx
	Micro-engine (22f) Registers (78, 80)	0000 0xxx xxxx xxxx
B000 XXXX	Micro-engine (22a) Registers (76b)	0010 1xxx xxxx xxxx
	Micro-engine (22b) Registers (76b)	1101 0xxx xxxx xxxx
	Micro-engine (22c) Registers (76b)	0001 1xxx xxxx xxxx
	Micro-engine (22d) Registers (76b)	0001 0xxx xxxx xxxx
	Micro-engine (22e) Registers (76b)	0000 1xxx xxxx xxxx
	Micro-engine (22f) Registers (76b)	0000 0xxx xxxx xxxx

FIG. 4

TEST AVAILABLE COPY

200 CPU 20 sends a WRITE command to the address space of the particular destination in the FBUS interface unit 28

202 Translation unit 30 latches the address and command type from the bus 32 and translates the address and WRITE command to a corresponding command in a format that simulates the format used by the pull engine 120

204 Translation unit 30 uses sideband command bus 134 to pass the translated command to a command interface 140 for the pull engine 120 (step 204)

206 Command interface 140 passes the translated WRITE command to the pull engine 120

208 Pull engine 120 executes the command

210 Pull engine 120 asserts control signal (wr\_to\_pull\_data) which is sent to the translation unit 30 via a control bus 136

212 Translation unit 30 promotes the WRITE data onto the bus 34A

214 Pull engine promotes the data to the FBUS interface destination indicated by the translated WRITE command

FIG. 5

220 CPU 20 sends a WRITE command to the address space of a particular register in one of micro-engines 22

222 Translation unit 30 latches the address and command type from the bus 32 and translates the address and the WRITE command to a corresponding command in a format used by the push engine 122

224 Translation unit 30 uses the sideband command bus 134 to pass the translated command to the command interface 144 for the push engine 122

226 Command interface 144 passes the translated command to the push engine 122

228 Push engine 122 executes the command

230 Push engine 122 asserts control signal (wr\_to\_push\_data)

232 Push engine 122 asserts address signals on address bus (Sbus\_push\_addr) 34C to enable the micro-engine 22 specified by the original WRITE command to accept the data on the Sbus\_push\_data bus 34B

FIG. 6

BEST AVAILABLE COPY

240 CPU 20 sends a READ command to the address space of the particular destination in the FBUS interface

242 Translation unit 30 latches the address and command type from the bus 32 and translates the address and READ command to a corresponding command in a format that simulates the format used by the push engine 122

244 Translation unit 30 uses the sideband command bus 134 to pass the translated command to the command interface 144 which passes the translated command to the push engine

246 Push engine 122 executes the READ command to place the data from the FBUS interface destination that was specified in the READ command onto the Sbus-Push\_data bus 34B

248 Push engine 122 asserts control signal (rd\_from\_push\_data) on the bus 136

250 Translation unit 30 promotes the data from the bus 34B to the core processor bus 32 so that the data can be received by the CPU 20

FIG. 7

260 CPU 20 sends a READ command to the address space of the particular register in one of the micro-engines 22

262 Translation unit 30 latches the address and command type from the bus 23 and translates the address and the READ command to a corresponding command in a format used by the pull engine 120

264 Translation unit 30 uses the sideband command bus 134 to pass the translated command to the command interface 140

266 Command interface 140 passes the translated READ command to the pull engine 120

268 Pull engine 120 executes the command so that the data from the micro-engine register specified in the READ command is placed on the Sbus\_pull\_data bus 34A

270 Pull engine 120 asserts control signal (rd\_from\_pull\_data) which is sent to the translation unit 30 via the control bus 136

272 Translation unit 30 promotes the data from the bus 34A to the core processor bus 32 so that the data can be received by the CPU 20

FIG. 8